Self Intro:

Hello, My name is XXX, and I've worked as a Full-Stack .NET Developer for 5 years. My last project was with Restaurant 365, a cloud-based provider, regarding their recipe tracking feature. The technologies I am familiar with for backend development are C#, ASP.NET Core, Entity Framework, and ADO.NET. I have proficiency in dealing with relational databases, including MS SQL Server and PostgreSQL. For Frontend, I have experience with HTML, CSS3, Angular 2+, and Razor views. As for DevOps tools, I have been using Git for source code management, Agile methodologies for project management, Postman, xUnit, Moq for testing, and RabbitMQ for messaging. So this is pretty much about myself, is there anything that you would like me to elaborate more?

Restaurant 365:
The company is called Restaurant 365, and it is a cloud-based restaurant enterprise management platform that provides centralized solutions for restaurants, by integrating the different operations in a restaurant, which results in the more efficient control of the business. So basically, my team and I were responsible for developing and maintaining the application overall, and I was involved in developing the recipe tracker feature, for that, I collaborated in the development of the inventory management module, recipe tracking module, recipe analysis module and the front end development for the users. This is a RESTful web application and some of the tech stacks we used were: C#, ASP.NET Core, Entity Framework, MS SQL Server, xUnit, Moq, Postman, AzureDevops, Angular 6, Git.

How our features work is that, first of all, the user is going to input the recipes for all the different menus from the recipe ingredients page, which basically includes the ingredients, the quantity of the ingredients, and recipe instructions. The recipes will then be sent to the recipe module using the httpClient and stored on the Recipe table through the EntityFramework. The main feature here is the reporting module, as it will provide the user with insights of their inventory as well as efficiency. How we do this is that we used data from the order module as well as the inventory module by calling a web api to their service. The Report module will call these APIs and generate report of the Actual vs. theoretical usage of the ingredients. Then based on the clients setup for the trigger, they are going to send an event to the Rabbit MQ. The report module will listen to the RabbitMQ and will process the report based on the event information. The report module will also provide search functionality meaning that it allows the user to view details of the report based on the date range using LINQ. In the report, there will be the quantity usage as well as dollar usage of the user’s restaurant, as well as the efficiency of the restaurant based on usage and spending.